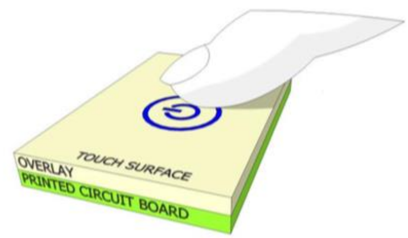
Exhibit 7

U.S. Patent No. 8,432,173 ("'173 Patent")

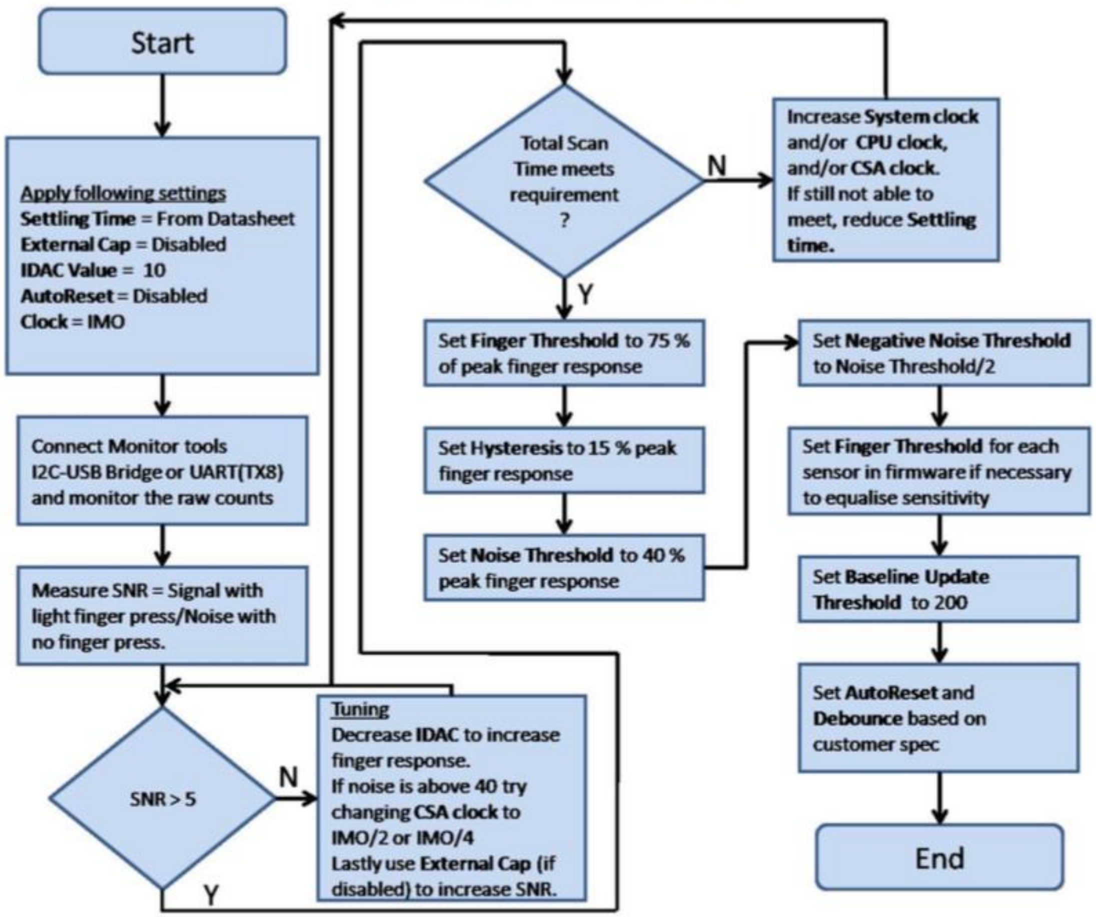
Exemplary Accused Product

Cypress products, including at least each of the following products (and their variations) infringe at least Claim 1 of the '173 Patent: Capsense enabled Cypress products, including MBR3, CY8CMBR2110, CY8CMBR2044, CY8CMBR2016, CY8CMBR2010, CY8CMBR3XXX, and Capsense-enabled PSoC. The infringement chart below is based on the Cypress CY8CMBR3106 with CapSense ("CapSense"), which is exemplary of the infringement of the '173 Patent.

Claim	CapSense
[1pre] A method comprising:	The Capsense touchcontroller provides capacitive touch sensing functionality. Cypress' CapSense controllers use changes in capacitance to detect the presence of a finger on or near a touch surface, as shown in Figure 2-1. This touch button example illustrates a capacitive sensor replacing a mechanical button. The sensing function is achieved using a combination of hardware and firmware. See the Glossary for the definitions of CapSense terms. Figure 2-1. Illustration of a Capacitive Sensor Application  *See* Getting Started with CapSense, at p. 10, https://www.cypress.com/file/41076/download Firmware is a vital component of the CapSense system. It processes the raw count data and makes logical decisions. The amount of firmware development required for your application depends on which CapSense controller family you select. *See* Getting Started with CapSense, at p. 12, https://www.cypress.com/file/41076/download

	Optimal CapSense system performance depends on the board layout, button dimensions, overlay material, and application requirements. In addition to these factors, switching frequency and threshold levels must be carefully selected for robust and reliable performance. Tuning is the process of determining the optimum values for these parameters. Tuning is required to maintain high sensitivity to touch and to compensate for process variations in the sensor board, overlay material, and environmental conditions. Many of the CapSense devices support SmartSense, Cypress' Auto-tuning algorithm, which automatically sets parameters for optimal performance and continuously compensates for system, manufacturing and environmental changes. See SmartSense Auto-Tuning for more information. *See* Getting Started with CapSense, at p. 18, https://www.cypress.com/file/41076/download PSoC uses Cypress patented capacitive touch sensing methods known as CapSense Sigma Delta (CSD) for self-capacitance sensing and CapSense Crosspoint (CSX) for mutual-capacitance scanning. The CSD and CSX touch sensing methods provide the industry's best-in-class Signal-to-Noise Ratio. These sensing methods are a combination of hardware and firmware techniques. *See* PSoC 4 and PSoC 6 MCU CapSense Design Guide, at p. 15, https://www.cypress.com/file/46081/download Tuning the touch sensing user interface is a critical step in ensuring proper system operation and a pleasant user experience. The typical design flow involves tuning the sensor interface in the initial design phase, during system integration, and finally production fine-tuning before the production ramp. Because tuning is an iterative process, it can be time-consuming. SmartSense Auto-Tuning helps to simplify the user interface development cycle. In addition, the method is easy to use and reduces the design cycle time by eliminating the tuning process throughout the product development cycle, from prototype to mass production. *See* Getting Started with CapSense, at p. 20, https://www.cypress.com/file/41076/download

Figure 8. CSA Calibration Flowchart



See CSA Software Filters with Ezl2Cs Slave on CY8C20xx6, at p. 10, https://www.cypress.com/file/137541/download

[1a] receiving one or more first signals indicating one or more first capacitive couplings of an object with a sensing element that comprises a sensing path that comprises a length, the first capacitive couplings corresponding to the object coming into proximity with the sensing element at a first position along the sensing path of the sensing element	The Capsense touchcontroller receives one or more first signals indicating one or more first capacitive couplings of an object with a sensing element that comprises a sensing path that comprises a length, the first capacitive couplings corresponding to the object coming into proximity with the sensing element at a first position along the sensing path of the sensing element. For example, the Capsense touchcontroller implements touch and movement functionality where the user places a finger/stylus on the connected touch sensor and moves it along a path or line. When the user's finger/stylus comes in contact with the touch sensor ("the object coming into proximity with the sensing element . . ."), one or capacitive couplings between the finger and the touch sensor is formed ("one or more capacitive couplings of an object with a sensing element . . ."). The Capsense touchcontroller receive one or more first signals indicating that the user places a finger/stylus on the touch sensor ("receiving one or more first signals indicating one or more first capacitive couplings of an object with a sensing element . . .").

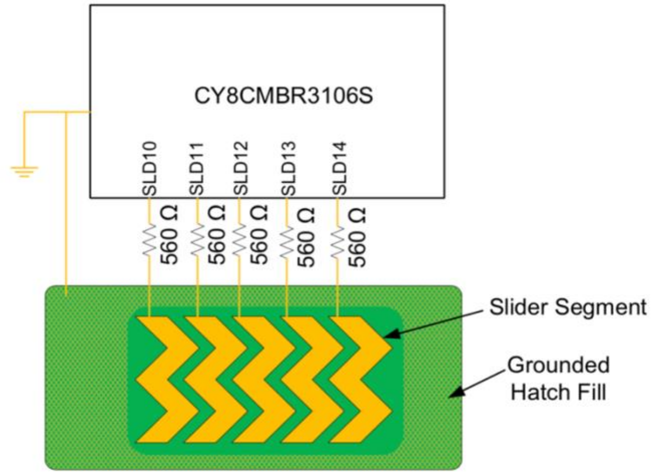
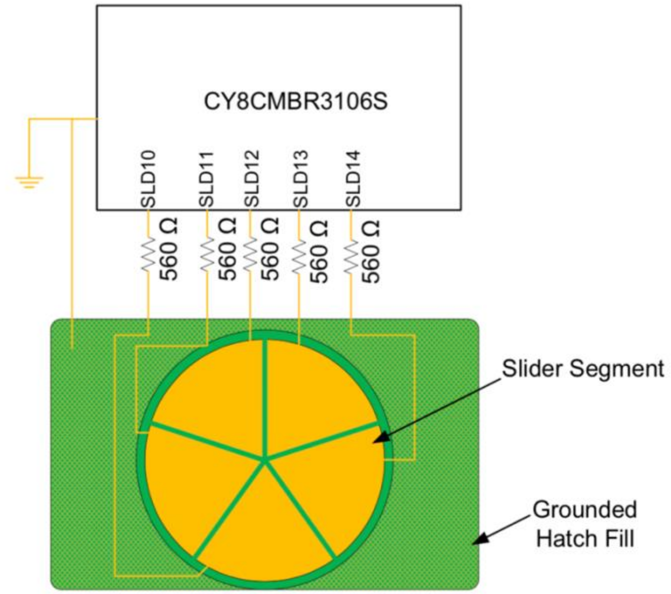
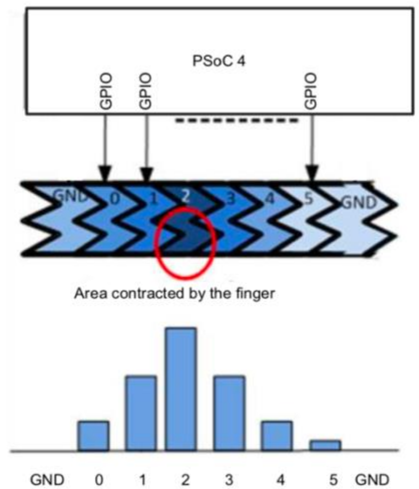
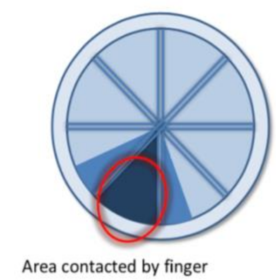
	Each slider segment connects to a pin marked for slider functionality on the CY8CMBR3106S controller. In CY8CMBR3106S, a SLDxx pin indicates that a slider segment can be connected to it. A zigzag pattern (double chevron) is recommended for slider segments. This layout ensures that when a segment is touched, the adjacent segments are also partially touched, which helps in the calculation of the centroid position. See the General Layout Guidelines section for layout recommendations. Figure 2-9. Linear Slider  Radial sliders are similar to linear sliders except that radial sliders are continuous. Figure 2-10 shows a typical radial slider. https://www.cypress.com/file/46366/download at 16.

	Figure 2-10. Radial Slider  https://www.cypress.com/file/46366/download at 16.

	2.7.2 Sliders (One-Dimensional) Sliders are used when the required input is in the form of a gradual increment or decrement. Examples include lighting control (dimmer), volume control, graphic equalizer, and speed control. Currently, the CapSense Component in PSoC Creator and ModusToolbox supports only self-capacitance-based sliders. Mutual capacitance-based sliders will be supported in future version of component. A slider consists of a one-dimensional array of capacitive sensors called segments, which are placed adjacent to one another. Touching one segment also results in partial activation of adjacent segments. The firmware processes the raw counts from the touched segment and the nearby segments to calculate the position of the geometric center of the finger touch, which is known as the **centroid position**. The actual resolution of the calculated centroid position is much higher than the number of segments in a slider. For example, a slider with five segments can resolve at least 100 physical finger positions. This high resolution gives smooth transitions of the centroid position as the finger glides across a slider. In a linear slider, the segments are arranged inline, as Figure 2-25 shows. Each slider segment connects to a PSoC GPIO. A zigzag pattern (double chevron) is recommended for slider segments. This layout ensures that when a segment is touched, the adjacent segments are also partially touched, which aids estimation of the centroid position. https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 26.

	Figure 2-25. Linear Slider  Radial sliders are similar to linear sliders except that radial sliders are continuous. Figure 2-26 shows a typical radial slider. Figure 2-26. Radial Slider  https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 27.

3.8.5 Slider Design

Figure 3-51 shows the recommended slider pattern for a linear slider and Table 3-10 shows the recommended values for each of the linear slider dimensions. Detailed explanation on the recommended layout guidelines are provided in the following sections.

Figure 3-51. Typical Linear Slider Pattern

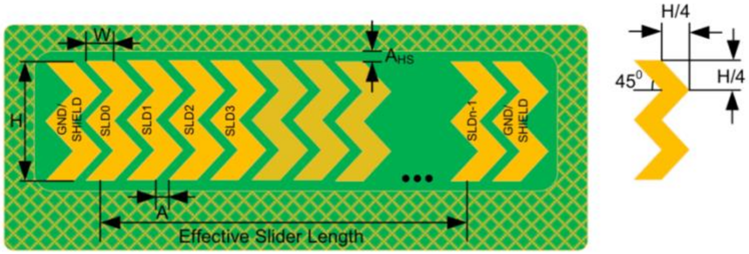


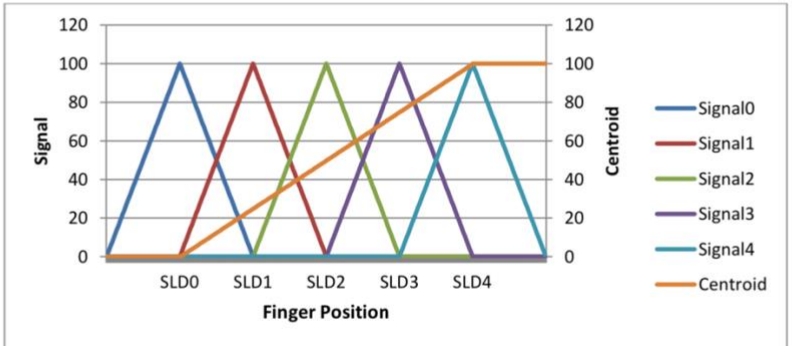
Table 3-10. Linear Slider Dimensions

Parameter	Acrylic Overlay Thickness	Minimum	Maximum	Recommended
Width of the Segment (W)	1 mm	2 mm	-	8 mm[a]
	3 mm	4 mm	-	
	4 mm	6 mm	-	
Height of the Segment (H)	-	7 mm[b]	15 mm	12 mm
Air-gap between Segments (A)	-	0.5 mm	2 mm	0.5 mm
Air-gap between hatch and slider (A_{HS})	-	0.5 mm	2 mm	Equal to overlay thickness

https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 75.

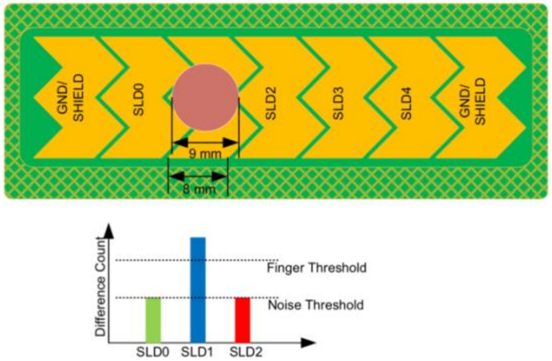
	Figure 3-52. Ideal Slider Segment Signals and Centroid Response

Figure 3-52. Ideal Slider Segment Signals and Centroid Response



Here, Signalx = Difference Count - Noise Threshold value for segment x.

Figure 3-53. Ideal Slider Signals



Equation 3-21. Segment width and air-gap relation with finger diameter

$$W + 2A = finger\ diameter$$

Typically, an average human finger diameter is approximately 9 mm. Based on this average finger diameter and Equation 3-21, the recommended slider-segment-width and air-gap is 8 mm and 0.5 mm respectively.

If the *slider-segment-width + 2 * air-gap* is lesser than *finger diameter*, as required per Equation 3-21, the centroid response will be non-linear. This is because, in this case, a finger placed on the slider will add capacitance, and hence valid signal to more than two slider-segments at some given position, as Figure 3-54 shows. Thus, calculated centroid position per Equation 3-21 will be non-linear, as Figure 3-54 shows.

https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 76.

[1b] determining based on one or more of the first signals the first position of the object along the sensing path;	The Capsense touchcontroller determines based on one or more of the first signals the first position of the object along the sensing path. *See, e.g.*, analysis and evidence in claim element 1[a] above. For example, at the beginning of a movement, the user's finger/stylus ("object") touches the touch sensor. The Capsense touchcontroller receives one or more first signals and determines the location where the user's finger/stylus touched the touch sensor ("the first position of the object along the sensing path"). For example, the Capsense touchcontroller has functionality that track touch and movement, including receiving information about the location, size, and movement of a touch occurring on the screen. This also includes information for the view or window in which the touch occurred, the location of the touch within the view or window, and the approximate radius of the touch. This also includes information about indicating when the touch occurred, and information about whether the touch began, moved, or ended.

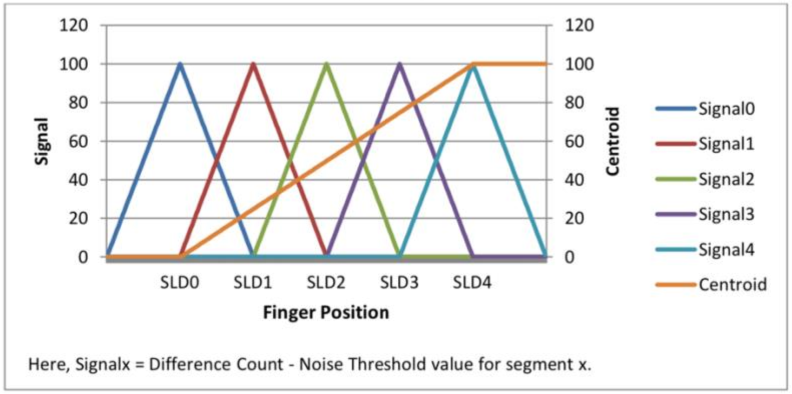
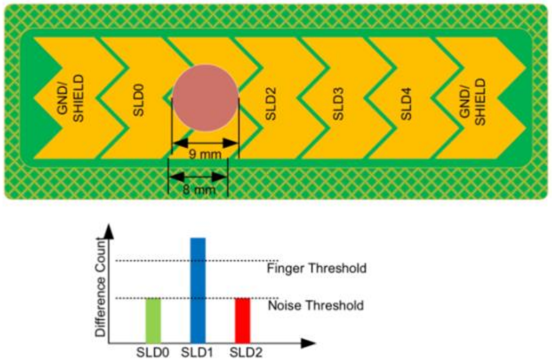
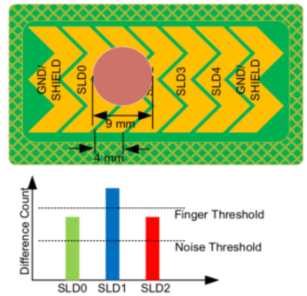
	Figure 3-52. Ideal Slider Segment Signals and Centroid Response  Figure 3-53. Ideal Slider Signals  Equation 3-21. Segment width and air-gap relation with finger diameter $W + 2A = finger\ diameter$ Typically, an average human finger diameter is approximately 9 mm. Based on this average finger diameter and Equation 3-21, the recommended slider-segment-width and air-gap is 8 mm and 0.5 mm respectively. If the *slider-segment-width + 2 * air-gap* is lesser than *finger diameter*, as required per Equation 3-21, the centroid response will be non-linear. This is because, in this case, a finger placed on the slider will add capacitance, and hence valid signal to more than two slider-segments at some given position, as Figure 3-54 shows. Thus, calculated centroid position per Equation 3-21 will be non-linear, as Figure 3-54 shows. https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 76.

Figure 3-54. Finger Causes Valid Signal on More Than Two Segments when Slider Segment Width Is Lower Than Recommended



Equation 3-22. Centroid algorithm used by CapSense

$$Centroid\ position = \left(\frac{S_{x+1} - S_{x-1}}{S_{x+1} + S_{x0} + S_{x-1}} + maximum\right) * \frac{Resolution}{(n-1)}$$

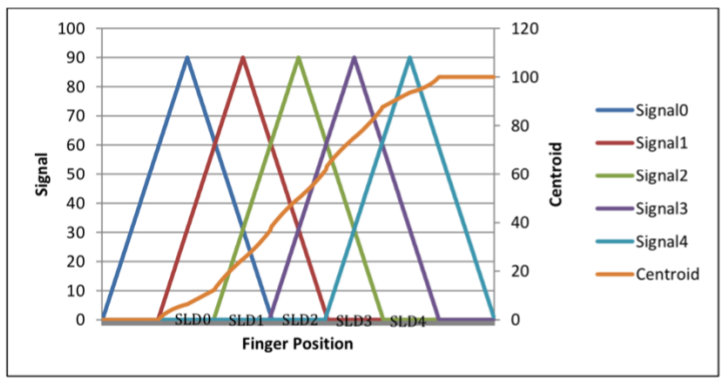
Resolution – API Resolution set in the Customizer,

n – Number of sensor elements in the Customizer.

maximum: Index of element which gives maximum signal.

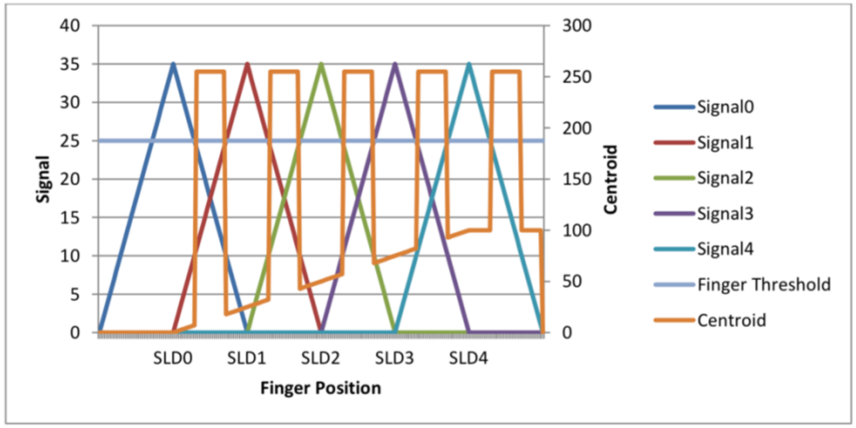
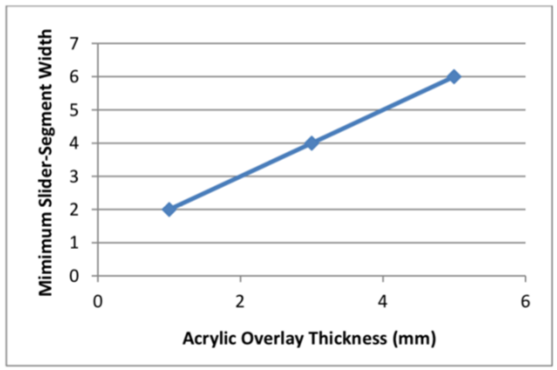
Si – different counts (with subtracted Noise Threshold value) near by the maximum position

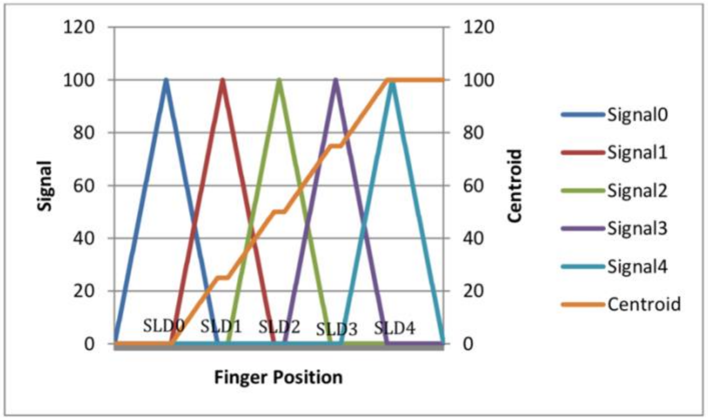
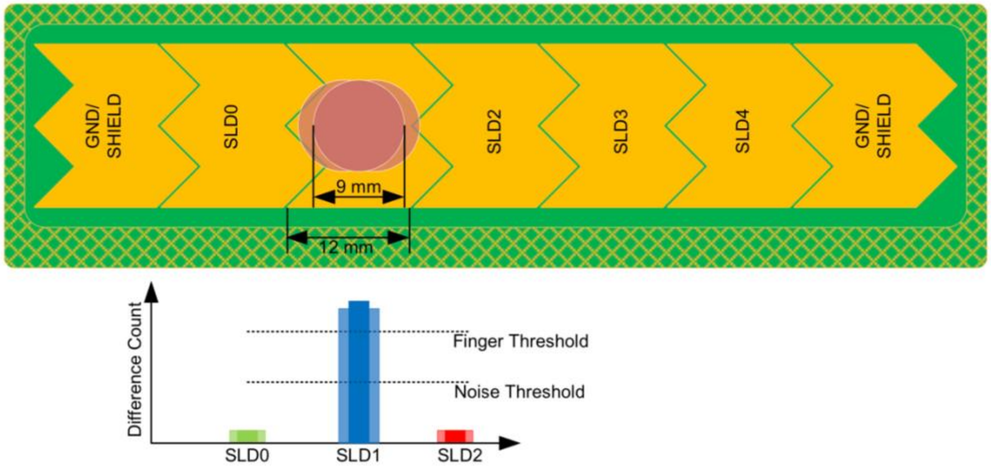
Figure 3-55. Nonlinear Centroid Response when Slider Segment Width Is Lower Than Recommended



Note that even though a *slider-segment-width* value of less than *finger diameter – 2 * air-gap* provides a non-linear centroid response, as Figure 3-54 shows; it may still be used in an end application where the linearity of reported centroid versus actual finger position does not play a significant role. However, a minimum value of slider-segment-width must be maintained, based on overlay thickness, such that, at any position on the effective slider length, at-least one slider-segment provides an SNR of >=5:1 (i.e. signal >= Finger Threshold parameter) at that position. If the slider-

https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 77.

	segment-width is too low, a finger may not be able to couple enough capacitance, and hence, none of the slider-segments will have a 5:1 SNR, resulting in a reported centroid value of 0xFF[8], as Figure 3-56 shows. Figure 3-56. Incorrect Centroid Reported when Slider Segment Width Is Too Low  The minimum value of slider-segment-width for certain specific overlay thickness values, for an acrylic overlay, are provided in Table 3-10. For acrylic overlays of thickness values, which are not specified in Table 3-10 and Figure 3-57 may be used to estimate the minimum slider-segment-width. Figure 3-57. Minimum Slider-Segment-Width w.r.t. Overlay Thickness for an Acrylic Overlay  If the *slider-segment-width + 2 * air-gap* is higher than *finger diameter*, as required per Equation 3-21, the centroid response will have flat spots i.e., if the finger is moved a little near the middle of any segment, the reported centroid position will remain constant as Figure 3-58 shows. This is because, as Figure 3-59 shows, when the finger is placed in the middle of a slider-segment, it will add valid signal only to that segment even if the finger is moved a little towards the adjacent segments.

	https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 78. Figure 3-58. Flat Spots (Nonresponsive Centroid) when Slider Segment Width Is Higher Than Recommended  Figure 3-59. Signal on Slider Segments when Slider Segment Width Is Higher Than Recommended  Note that if the *slider-segment-width + 2 * air-gap* is higher than *finger diameter,* it may be possible to increase and adjust the sensitivity of all the slider segments such that even if the finger is placed in middle of a slider-segment, the adjacent sensors report a difference count value equal to noise threshold value (as required per

	https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 79.

2.3.1 CapSense Sigma Delta Modulator (CSD) Sensing Method

Figure 2-11 shows a simplified block diagram of the CSD method.

In CSD, each GPIO has a switched-capacitance circuit that converts the sensor capacitance into an equivalent current. An analog multiplexer then selects one of the currents and feeds it into the current to digital converter. The current to digital converter is similar to a sigma delta ADC. The output count of the current to digital converter, known as **raw count**, is a digital value that is proportional to the self-capacitance between the electrodes.

Equation 2-3. Raw Count and Sensor Capacitance Relationship in CSD

$$\text{raw count} = G_C C_S$$

Where G_C is the capacitance to digital conversion gain of CSD, and

C_S is the self-capacitance of the electrode.

Figure 2-11. Simplified Diagram of CapSense Sigma Delta Method

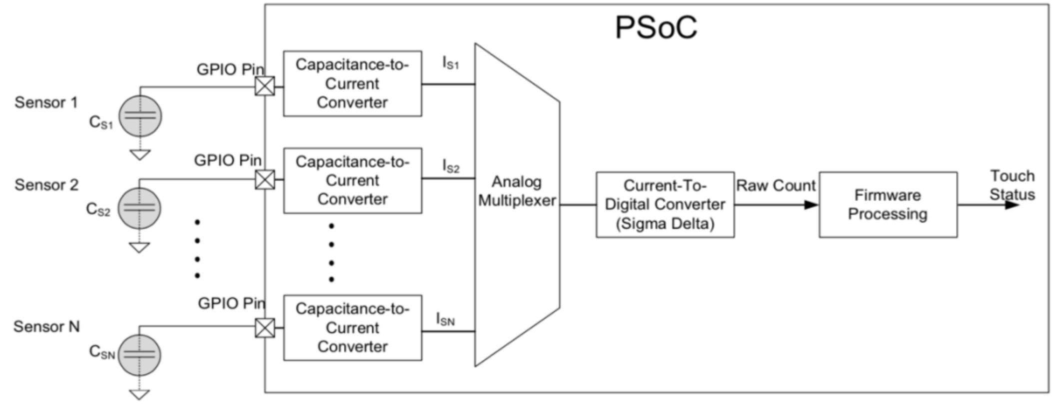
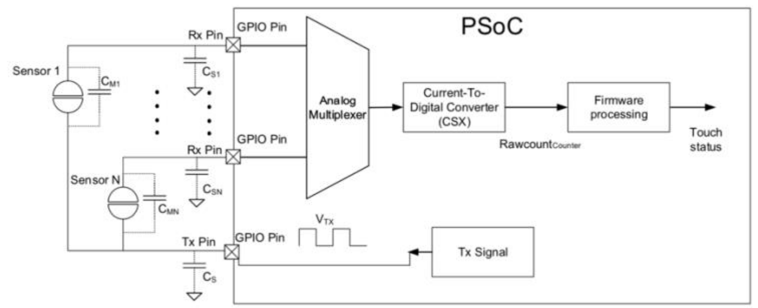
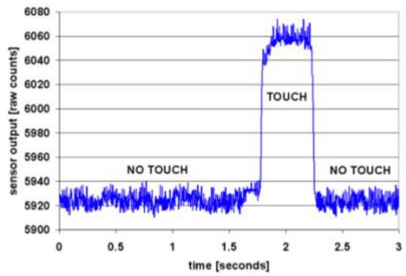


Figure 2-13 shows a plot of raw count over time. When a finger touches the sensor, the C_S increases from C_P to $C_P + C_F$, and the raw count increases. By comparing the change in raw count to a predetermined threshold, logic in firmware decides whether the sensor is active (finger is present).

https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 15.

	2.3.2 CapSense Crosspoint (CSX) Sensing Method Figure 2-12 shows the simplified block diagram of the CSX method. Figure 2-12. Simplified Diagram of CSX Method  With CSX, a voltage on the Tx pin (or Tx electrode) couples charge on to the RX pin. This charge is proportional to the mutual capacitance between the Tx and Rx electrodes. An analog multiplexer then selects one of the Rx channel and feeds it into the current to digital converter. The output count of the current to digital converter, known as $\textbf{Rawcount}_{\textbf{Counter}}$, is a digital value that is proportional to the mutual-capacitance between the Rx and Tx electrodes as shown by Equation 2-4. Equation 2-4. Raw Count and Sensor Capacitance Relationship in CSX $\text{Rawcount}_{\text{Counter}} = G_{CM}\, C_M$ Where G_{CM} is the capacitance to digital conversion gain of Mutual Capacitance method, and C_M is the mutual-capacitance between two electrodes. Figure 2-13 shows a plot of raw count over time. When a finger touches the sensor, C_M decreases from C_M to C^1_M (see Figure 2-10) hence the counter output decreases. The firmware normalizes the raw count such that the raw counts go high when C_M decreases. This is to maintain the same visual representation of raw count between CSD and CSX methods. By comparing the change in raw count to a predetermined threshold, logic in firmware decides whether the sensor is active (finger is present). Figure 2-13. Raw Count versus Time  https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 16.

[1c] setting a parameter to an initial value based on the first position of the object along the sensing path, the initial value comprising a particular parameter value and being associated with a range of parameter values, the range of parameter values being associated with the length of the sensing path;	The Capsense touchcontroller sets a parameter to an initial value based on the first position of the object along the sensing path, the initial value comprising a particular parameter value and being associated with a range of parameter values, the range of parameter values being associated with the length of the sensing path. The Capsense touchcontroller sets a parameter to an initial value based on the first position of the object along the sensing path. In the Capsense touchcontroller, the initial value comprises a particular parameter value and being associated with a range of parameter values, the range of parameter values being associated with the length of the sensing path. For example, the initial value comprises a particular parameter value or setting that can be adjusted, and is associated with a range of parameter values, e.g., the range of adjustment. Further, the range of parameter values or settings is associated with the length of a sensing path and can be adjusted along the length of that path.

Figure 3-52. Ideal Slider Segment Signals and Centroid Response

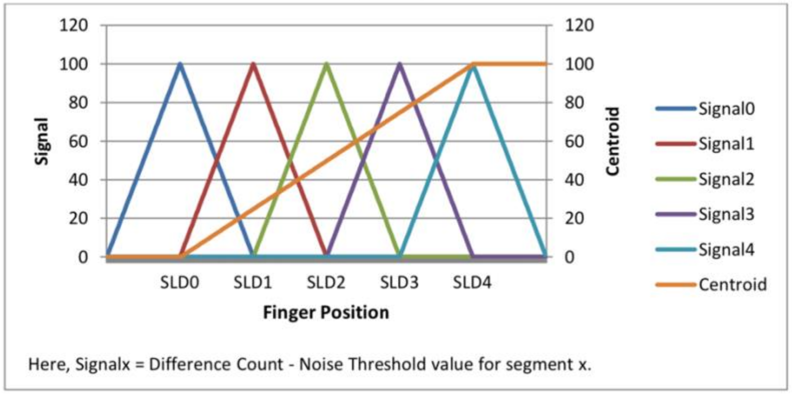
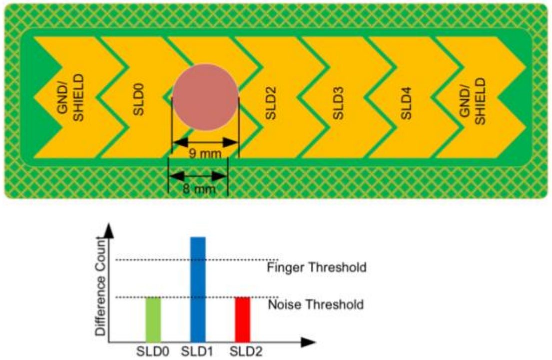


Figure 3-53. Ideal Slider Signals



Equation 3-21. Segment width and air-gap relation with finger diameter

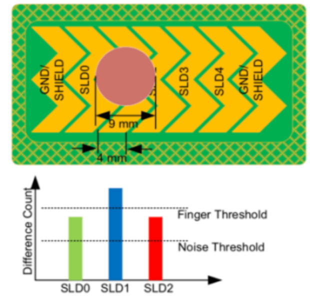
$$W + 2A = finger\ diameter$$

Typically, an average human finger diameter is approximately 9 mm. Based on this average finger diameter and Equation 3-21, the recommended slider-segment-width and air-gap is 8 mm and 0.5 mm respectively.

If the *slider-segment-width + 2 * air-gap* is lesser than *finger diameter*, as required per Equation 3-21, the centroid response will be non-linear. This is because, in this case, a finger placed on the slider will add capacitance, and hence valid signal to more than two slider-segments at some given position, as Figure 3-54 shows. Thus, calculated centroid position per Equation 3-21 will be non-linear, as Figure 3-54 shows.

https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 76.

Figure 3-54. Finger Causes Valid Signal on More Than Two Segments when Slider Segment Width Is Lower Than Recommended



Equation 3-22. Centroid algorithm used by CapSense

$$Centroid\ position = \left(\frac{S_{x+1} - S_{x-1}}{S_{x+1} + S_{x0} + S_{x-1}} + maximum\right) * \frac{Resolution}{(n-1)}$$

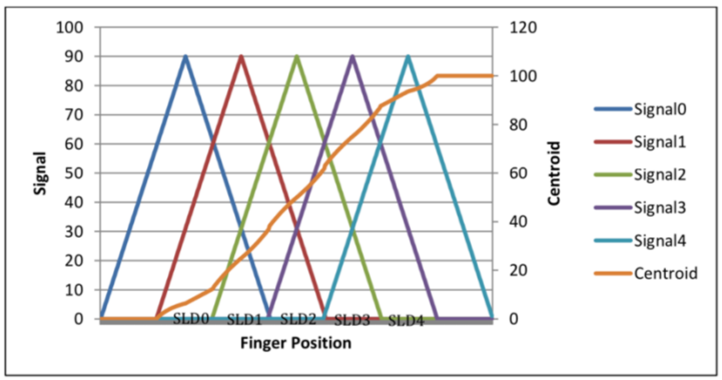
Resolution – API Resolution set in the Customizer,

n – Number of sensor elements in the Customizer.

maximum: Index of element which gives maximum signal.

Si – different counts (with subtracted Noise Threshold value) near by the maximum position

Figure 3-55. Nonlinear Centroid Response when Slider Segment Width Is Lower Than Recommended

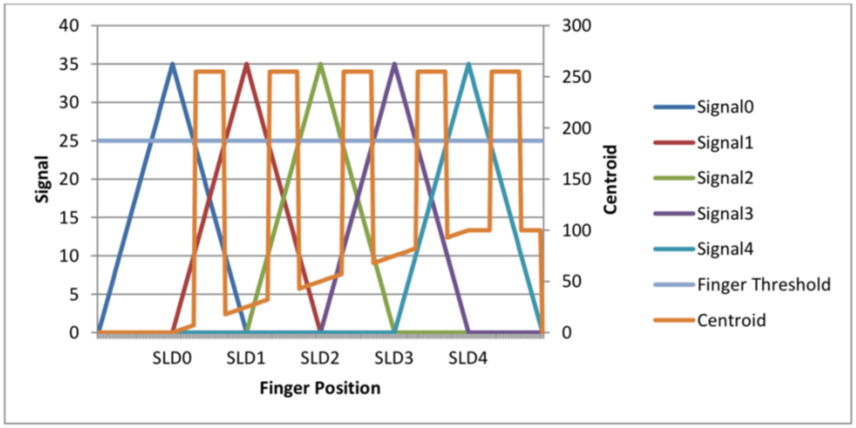


Note that even though a *slider-segment-width* value of less than *finger diameter – 2 * air-gap* provides a non-linear centroid response, as Figure 3-54 shows; it may still be used in an end application where the linearity of reported centroid versus actual finger position does not play a significant role. However, a minimum value of slider-segment-width must be maintained, based on overlay thickness, such that, at any position on the effective slider length, at-least one slider-segment provides an SNR of >=5:1 (i.e. signal >= Finger Threshold parameter) at that position. If the slider-

https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 77.

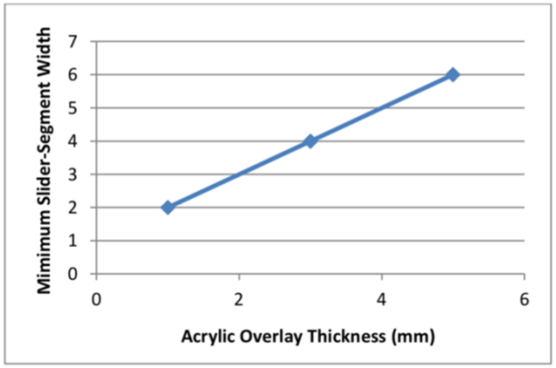
segment-width is too low, a finger may not be able to couple enough capacitance, and hence, none of the slider-segments will have a 5:1 SNR, resulting in a reported centroid value of 0xFF[8], as Figure 3-56 shows.

Figure 3-56. Incorrect Centroid Reported when Slider Segment Width Is Too Low



The minimum value of slider-segment-width for certain specific overlay thickness values, for an acrylic overlay, are provided in Table 3-10. For acrylic overlays of thickness values, which are not specified in Table 3-10 and Figure 3-57 may be used to estimate the minimum slider-segment-width.

Figure 3-57. Minimum Slider-Segment-Width w.r.t. Overlay Thickness for an Acrylic Overlay



If the *slider-segment-width + 2 *air-gap* is higher than *finger diameter*, as required per Equation 3-21, the centroid response will have flat spots i.e., if the finger is moved a little near the middle of any segment, the reported centroid position will remain constant as Figure 3-58 shows. This is because, as Figure 3-59 shows, when the finger is placed in the middle of a slider-segment, it will add valid signal only to that segment even if the finger is moved a little towards the adjacent segments.

	https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 78.

Figure 3-58. Flat Spots (Nonresponsive Centroid) when Slider Segment Width Is Higher Than Recommended

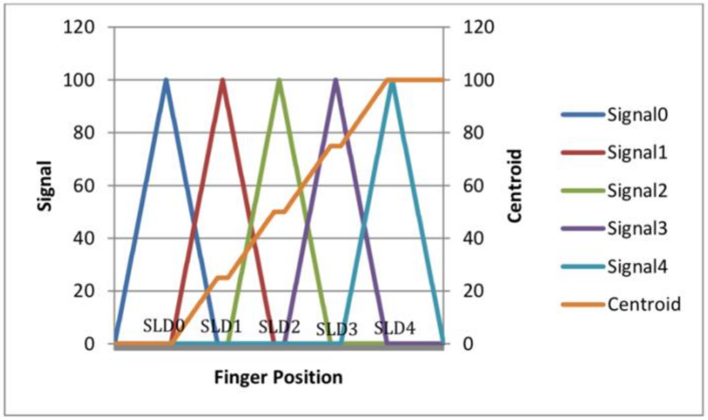
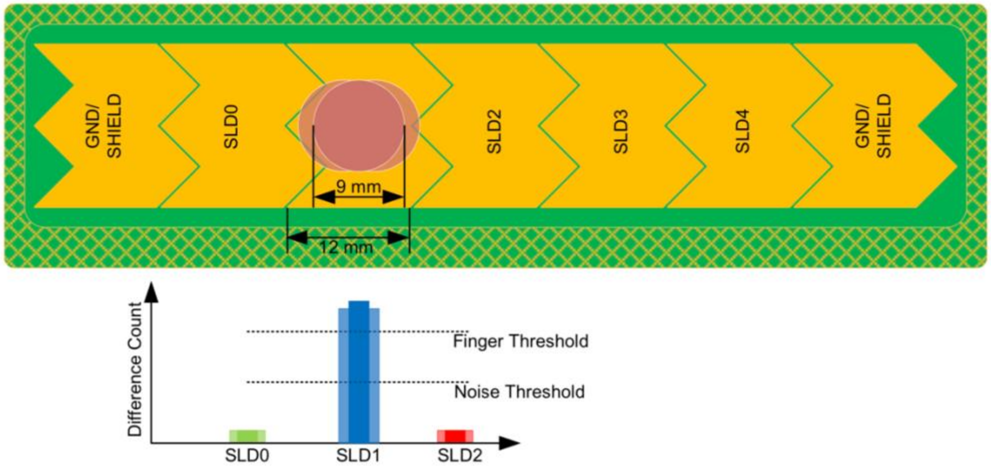
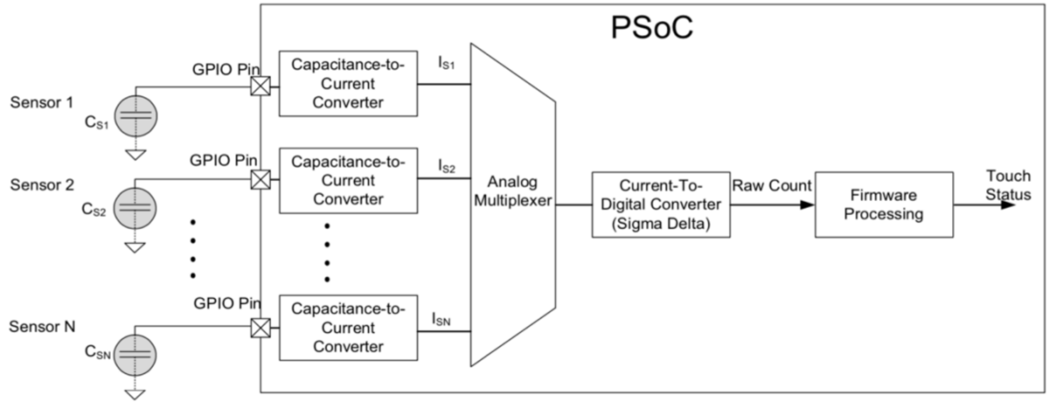


Figure 3-59. Signal on Slider Segments when Slider Segment Width Is Higher Than Recommended



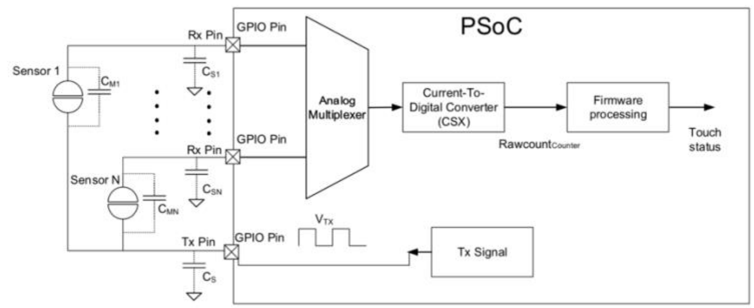
Note that if the *slider-segment-width + 2 * air-gap* is higher than *finger diameter*, it may be possible to increase and adjust the sensitivity of all the slider segments such that even if the finger is placed in middle of a slider-segment, the adjacent sensors report a difference count value equal to noise threshold value (as required per

	https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 79. 2.3.1 CapSense Sigma Delta Modulator (CSD) Sensing Method Figure 2-11 shows a simplified block diagram of the CSD method. In CSD, each GPIO has a switched-capacitance circuit that converts the sensor capacitance into an equivalent current. An analog multiplexer then selects one of the currents and feeds it into the current to digital converter. The current to digital converter is similar to a sigma delta ADC. The output count of the current to digital converter, known as **raw count**, is a digital value that is proportional to the self-capacitance between the electrodes. Equation 2-3. Raw Count and Sensor Capacitance Relationship in CSD $\text{raw count} = G_C C_S$ Where G_C is the capacitance to digital conversion gain of CSD, and C_S is the self-capacitance of the electrode. Figure 2-11. Simplified Diagram of CapSense Sigma Delta Method  Figure 2-13 shows a plot of raw count over time. When a finger touches the sensor, the C_S increases from C_P to $C_P + C_F$, and the raw count increases. By comparing the change in raw count to a predetermined threshold, logic in firmware decides whether the sensor is active (finger is present). https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 15.

2.3.2 CapSense Crosspoint (CSX) Sensing Method

Figure 2-12 shows the simplified block diagram of the CSX method.

Figure 2-12. Simplified Diagram of CSX Method



With CSX, a voltage on the Tx pin (or Tx electrode) couples charge on to the RX pin. This charge is proportional to the mutual capacitance between the Tx and Rx electrodes. An analog multiplexer then selects one of the Rx channel and feeds it into the current to digital converter.

The output count of the current to digital converter, known as $\mathbf{Rawcount_{Counter}}$, is a digital value that is proportional to the mutual-capacitance between the Rx and Tx electrodes as shown by Equation 2-4.

Equation 2-4. Raw Count and Sensor Capacitance Relationship in CSX

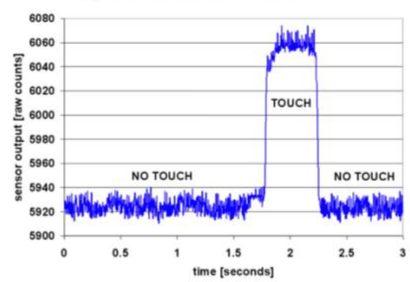
$$\mathrm{Rawcount_{Counter}} = G_{CM}\, C_M$$

Where G_{CM} is the capacitance to digital conversion gain of Mutual Capacitance method, and

C_M is the mutual-capacitance between two electrodes.

Figure 2-13 shows a plot of raw count over time. When a finger touches the sensor, C_M decreases from C_M to C^1_M (see Figure 2-10) hence the counter output decreases. The firmware normalizes the raw count such that the raw counts go high when C_M decreases. This is to maintain the same visual representation of raw count between CSD and CSX methods. By comparing the change in raw count to a predetermined threshold, logic in firmware decides whether the sensor is active (finger is present).

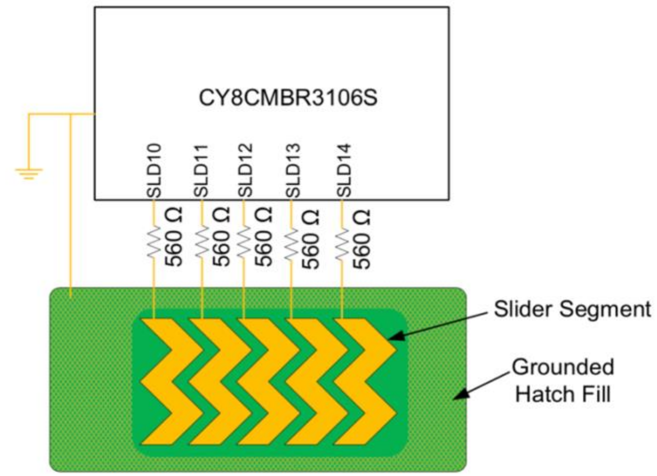
Figure 2-13. Raw Count versus Time



https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 16.

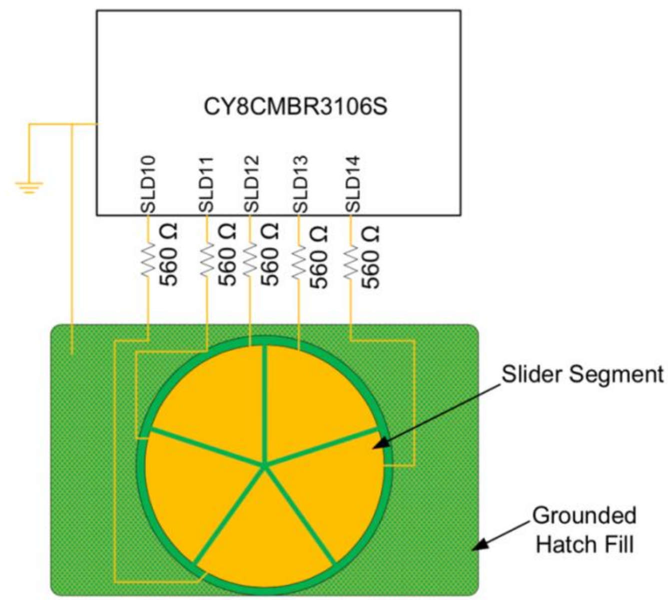
Each slider segment connects to a pin marked for slider functionality on the CY8CMBR3106S controller. In CY8CMBR3106S, a SLDxx pin indicates that a slider segment can be connected to it. A zigzag pattern (double chevron) is recommended for slider segments. This layout ensures that when a segment is touched, the adjacent segments are also partially touched, which helps in the calculation of the centroid position. See the General Layout Guidelines section for layout recommendations.

Figure 2-9. Linear Slider



Radial sliders are similar to linear sliders except that radial sliders are continuous. Figure 2-10 shows a typical radial slider.

https://www.cypress.com/file/46366/download at 16.

	Figure 2-10. Radial Slider  https://www.cypress.com/file/46366/download at 16.
[1d] receiving one or more second signals indicating one or more second capacitive couplings of the object with the sensing element, the second capacitive couplings corresponding to a displacement of the object along the sensing path from the first position; and	The Capsense touchcontroller receives one or more second signals indicating one or more second capacitive couplings of the object with the sensing element, the second capacitive couplings corresponding to a displacement of the object along the sensing path from the first position. *See, e.g.*, analysis and evidence in claim element 1[a] above. For example, the Capsense touchcontroller implements touch and movement functionality where the user places a finger/stylus on the touch sensor and moves it along a path or line. When the user's finger/stylus moves along a path a line ("displacement of the object along the sensing path from the first position"), one or more capacitive couplings between the user's finger/stylus and the touch sensor are formed ("the second capacitive couplings corresponding to a displacement . . ."). Further, the Capsense touchcontroller receives one or more signals indicating the

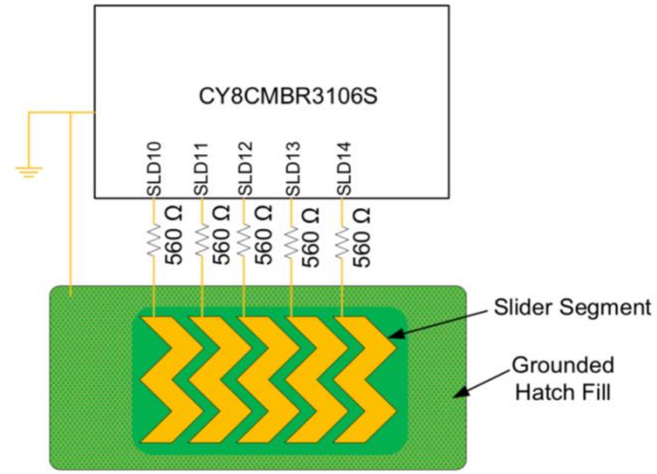
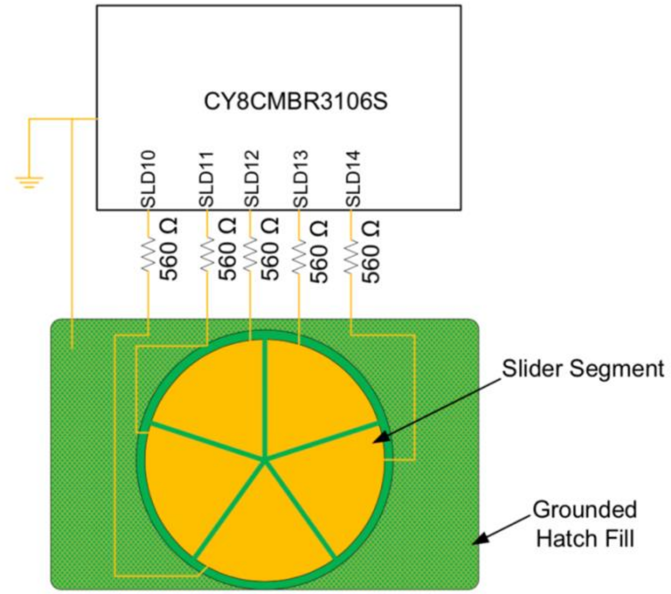
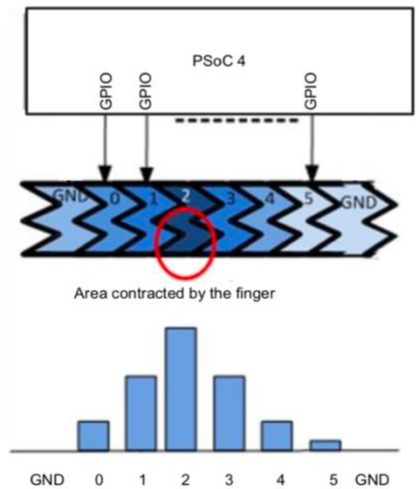
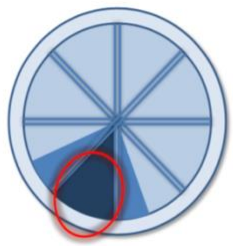
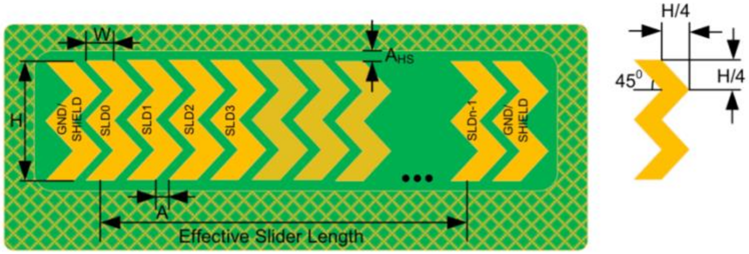
	movement ("receiving one or more second signals indicating one or more second capacitive couplings of the object with the sensing element"). Each slider segment connects to a pin marked for slider functionality on the CY8CMBR3106S controller. In CY8CMBR3106S, a SLDxx pin indicates that a slider segment can be connected to it. A zigzag pattern (double chevron) is recommended for slider segments. This layout ensures that when a segment is touched, the adjacent segments are also partially touched, which helps in the calculation of the centroid position. See the General Layout Guidelines section for layout recommendations. Figure 2-9. Linear Slider  Radial sliders are similar to linear sliders except that radial sliders are continuous. Figure 2-10 shows a typical radial slider. https://www.cypress.com/file/46366/download at 16.

	Figure 2-10. Radial Slider  https://www.cypress.com/file/46366/download at 16.

	2.7.2 Sliders (One-Dimensional) Sliders are used when the required input is in the form of a gradual increment or decrement. Examples include lighting control (dimmer), volume control, graphic equalizer, and speed control. Currently, the CapSense Component in PSoC Creator and ModusToolbox supports only self-capacitance-based sliders. Mutual capacitance-based sliders will be supported in future version of component. A slider consists of a one-dimensional array of capacitive sensors called segments, which are placed adjacent to one another. Touching one segment also results in partial activation of adjacent segments. The firmware processes the raw counts from the touched segment and the nearby segments to calculate the position of the geometric center of the finger touch, which is known as the **centroid position**. The actual resolution of the calculated centroid position is much higher than the number of segments in a slider. For example, a slider with five segments can resolve at least 100 physical finger positions. This high resolution gives smooth transitions of the centroid position as the finger glides across a slider. In a linear slider, the segments are arranged inline, as Figure 2-25 shows. Each slider segment connects to a PSoC GPIO. A zigzag pattern (double chevron) is recommended for slider segments. This layout ensures that when a segment is touched, the adjacent segments are also partially touched, which aids estimation of the centroid position. https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 26.

	Figure 2-25. Linear Slider  Radial sliders are similar to linear sliders except that radial sliders are continuous. Figure 2-26 shows a typical radial slider. Figure 2-26. Radial Slider  https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 27.

	3.8.5 Slider Design Figure 3-51 shows the recommended slider pattern for a linear slider and Table 3-10 shows the recommended values for each of the linear slider dimensions. Detailed explanation on the recommended layout guidelines are provided in the following sections. Figure 3-51. Typical Linear Slider Pattern  Table 3-10. Linear Slider Dimensions

Parameter	Acrylic Overlay Thickness	Minimum	Maximum	Recommended
Width of the Segment (W)	1 mm	2 mm	-	8 mm[6]
	3 mm	4 mm	-	
	4 mm	6 mm	-	
Height of the Segment (H)	-	7 mm[b]	15 mm	12 mm
Air-gap between Segments (A)	-	0.5 mm	2 mm	0.5 mm
Air-gap between hatch and slider (A_{HS})	-	0.5 mm	2 mm	Equal to overlay thickness

https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 75.

Figure 3-52. Ideal Slider Segment Signals and Centroid Response

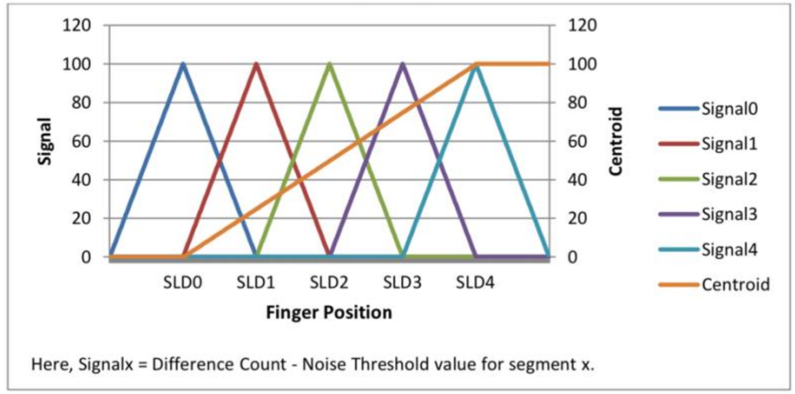
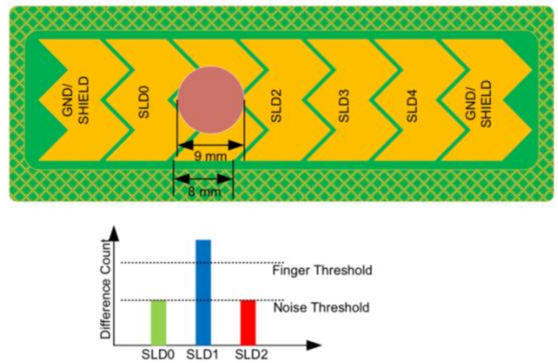


Figure 3-53. Ideal Slider Signals



Equation 3-21. Segment width and air-gap relation with finger diameter

$$W + 2A = finger\ diameter$$

Typically, an average human finger diameter is approximately 9 mm. Based on this average finger diameter and Equation 3-21, the recommended slider-segment-width and air-gap is 8 mm and 0.5 mm respectively.

If the *slider-segment-width + 2 * air-gap* is lesser than *finger diameter*, as required per Equation 3-21, the centroid response will be non-linear. This is because, in this case, a finger placed on the slider will add capacitance, and hence valid signal to more than two slider-segments at some given position, as Figure 3-54 shows. Thus, calculated centroid position per Equation 3-21 will be non-linear, as Figure 3-54 shows.

https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 76.

[1e] determining based on one or more of the second signals the displacement of the object along the sensing path; and	The Capsense touchcontroller determines based on one or more of the second signals the displacement of the object along the sensing path. *See, e.g.*, analysis and evidence in claim elements 1[a] and 1[b] above. For example, as the user's finger/stylus ("object") moves in a line or path on the touch sensor ("the displacement of the object along the sensing path"), the Capsense touchcontroller receives one or more second signals that indicates the movement. For example, the Capsense touchcontroller has functionality that tracks touch and movement, including receiving information about the location, size, and movement of a touch occurring on the screen. This also includes information for the view or window in which the touch occurred, the location of the touch within the view or window, and the approximate radius of the touch. This also includes information about indicating when the touch occurred, and information about whether the touch began, moved, or ended.
[1f] adjusting the parameter within the range of parameter values based on the displacement of the object along the sensing path.	The Capsense touchcontroller adjusts the parameter within the range of parameter values based on the displacement of the object along the sensing path. *See, e.g.*, analysis and evidence in claim element 1[c] and 1[d] above. For example, based on the movement of the user's finger/stylus along the sensing path, the Capsense touchcontroller adjusts the parameter within the range of parameter values. For example, the Capsense touchcontroller includes a particular parameter value or setting that can be adjusted, and is associated with a range of parameter values, i.e., the range of adjustment. Further, the range of parameter values or settings is associated with the length of a sensing path and can be adjusted along the length of that path.

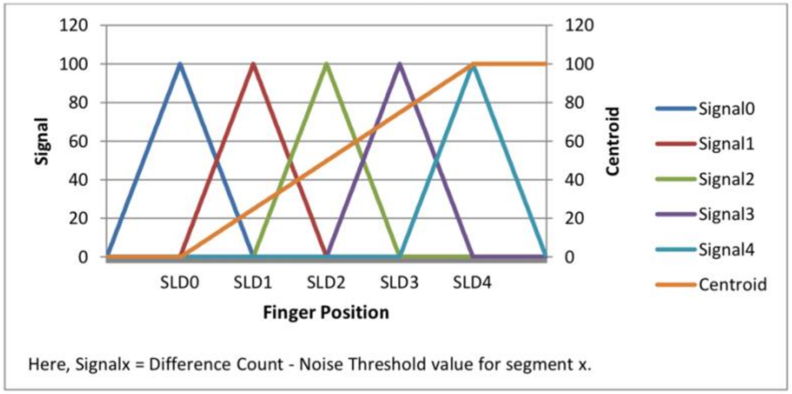
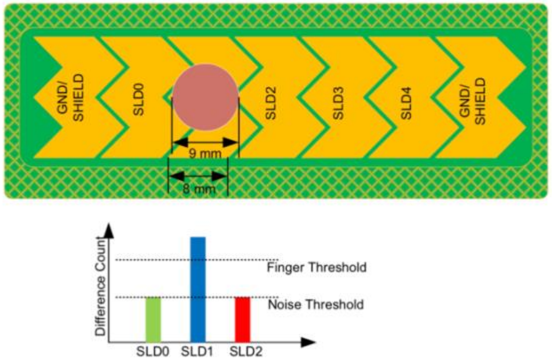
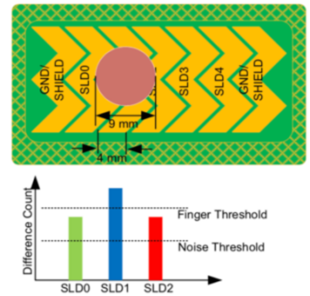
	Figure 3-52. Ideal Slider Segment Signals and Centroid Response  Figure 3-53. Ideal Slider Signals  Equation 3-21. Segment width and air-gap relation with finger diameter $W + 2A = finger\ diameter$ Typically, an average human finger diameter is approximately 9 mm. Based on this average finger diameter and Equation 3-21, the recommended slider-segment-width and air-gap is 8 mm and 0.5 mm respectively. If the *slider-segment-width + 2 * air-gap* is lesser than *finger diameter*, as required per Equation 3-21, the centroid response will be non-linear. This is because, in this case, a finger placed on the slider will add capacitance, and hence valid signal to more than two slider-segments at some given position, as Figure 3-54 shows. Thus, calculated centroid position per Equation 3-21 will be non-linear, as Figure 3-54 shows. https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 76.

Figure 3-54. Finger Causes Valid Signal on More Than Two Segments when Slider Segment Width Is Lower Than Recommended



Equation 3-22. Centroid algorithm used by CapSense

$$Centroid\ position = \left(\frac{S_{x+1} - S_{x-1}}{S_{x+1} + S_{x0} + S_{x-1}} + maximum\right) * \frac{Resolution}{(n-1)}$$

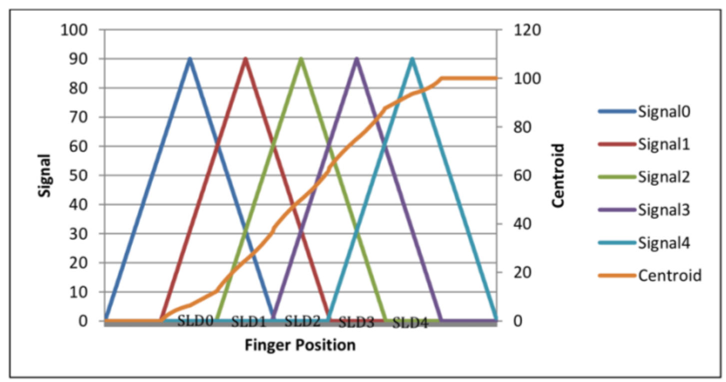
Resolution – API Resolution set in the Customizer,

n – Number of sensor elements in the Customizer.

maximum: Index of element which gives maximum signal.

Si – different counts (with subtracted Noise Threshold value) near by the maximum position

Figure 3-55. Nonlinear Centroid Response when Slider Segment Width Is Lower Than Recommended

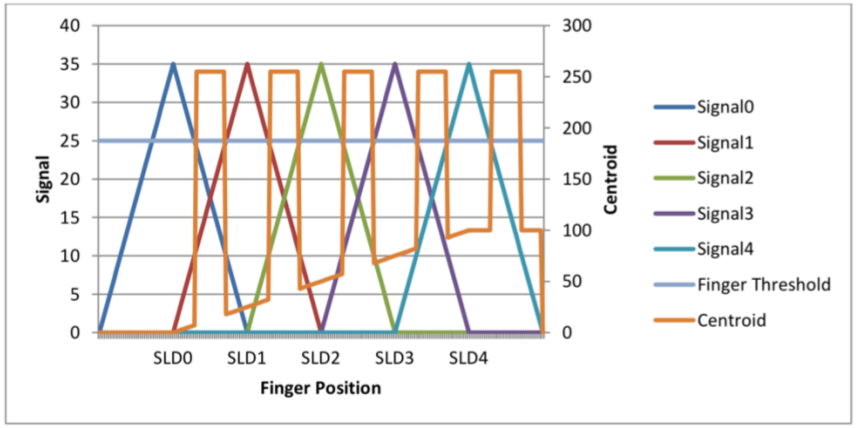


Note that even though a *slider-segment-width* value of less than *finger diameter – 2 * air-gap* provides a non-linear centroid response, as Figure 3-54 shows; it may still be used in an end application where the linearity of reported centroid versus actual finger position does not play a significant role. However, a minimum value of slider-segment-width must be maintained, based on overlay thickness, such that, at any position on the effective slider length, at-least one slider-segment provides an SNR of >=5:1 (i.e. signal >= Finger Threshold parameter) at that position. If the slider-

https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 77.

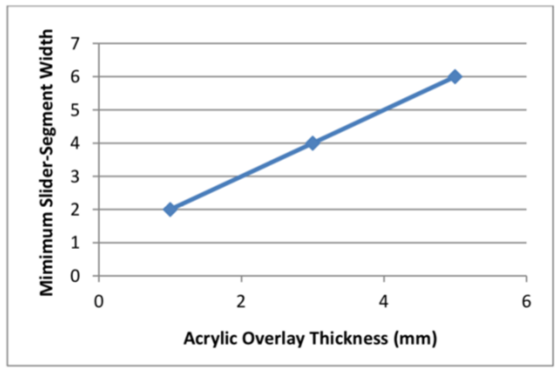
segment-width is too low, a finger may not be able to couple enough capacitance, and hence, none of the slider-segments will have a 5:1 SNR, resulting in a reported centroid value of 0xFF[8], as Figure 3-56 shows.

Figure 3-56. Incorrect Centroid Reported when Slider Segment Width Is Too Low

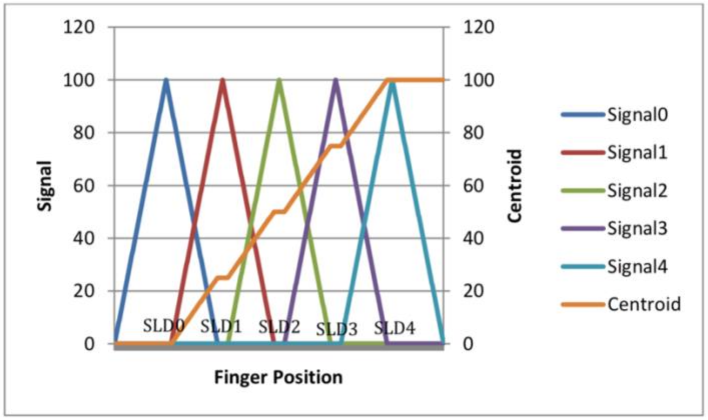
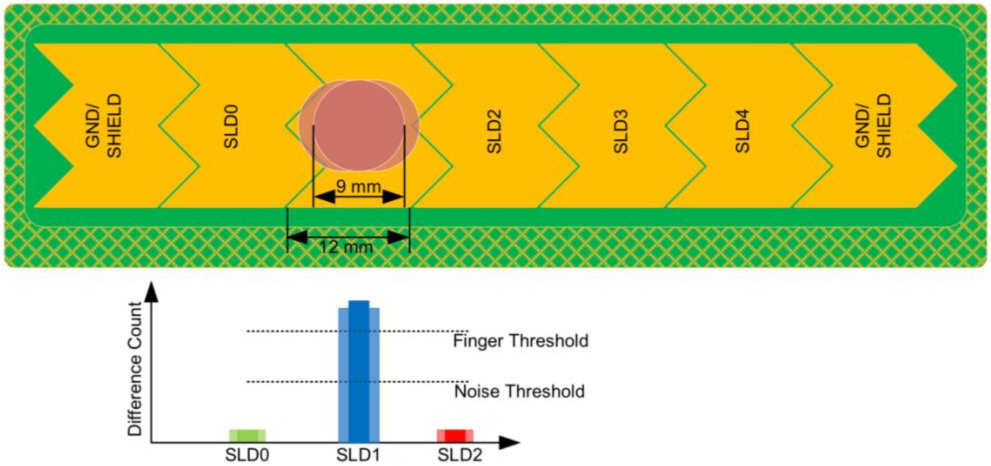


The minimum value of slider-segment-width for certain specific overlay thickness values, for an acrylic overlay, are provided in Table 3-10. For acrylic overlays of thickness values, which are not specified in Table 3-10 and Figure 3-57 may be used to estimate the minimum slider-segment-width.

Figure 3-57. Minimum Slider-Segment-Width w.r.t. Overlay Thickness for an Acrylic Overlay



If the *slider-segment-width + 2 * air-gap* is higher than *finger diameter*, as required per Equation 3-21, the centroid response will have flat spots i.e., if the finger is moved a little near the middle of any segment, the reported centroid position will remain constant as Figure 3-58 shows. This is because, as Figure 3-59 shows, when the finger is placed in the middle of a slider-segment, it will add valid signal only to that segment even if the finger is moved a little towards the adjacent segments.

	https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 78. Figure 3-58. Flat Spots (Nonresponsive Centroid) when Slider Segment Width Is Higher Than Recommended  Figure 3-59. Signal on Slider Segments when Slider Segment Width Is Higher Than Recommended  Note that if the *slider-segment-width + 2 * air-gap* is higher than *finger diameter*, it may be possible to increase and adjust the sensitivity of all the slider segments such that even if the finger is placed in middle of a slider-segment, the adjacent sensors report a difference count value equal to noise threshold value (as required per

	https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 79.

2.3.1 CapSense Sigma Delta Modulator (CSD) Sensing Method

Figure 2-11 shows a simplified block diagram of the CSD method.

In CSD, each GPIO has a switched-capacitance circuit that converts the sensor capacitance into an equivalent current. An analog multiplexer then selects one of the currents and feeds it into the current to digital converter. The current to digital converter is similar to a sigma delta ADC. The output count of the current to digital converter, known as **raw count**, is a digital value that is proportional to the self-capacitance between the electrodes.

Equation 2-3. Raw Count and Sensor Capacitance Relationship in CSD

$$\text{raw count} = G_C C_S$$

Where G_C is the capacitance to digital conversion gain of CSD, and

C_S is the self-capacitance of the electrode.

Figure 2-11. Simplified Diagram of CapSense Sigma Delta Method

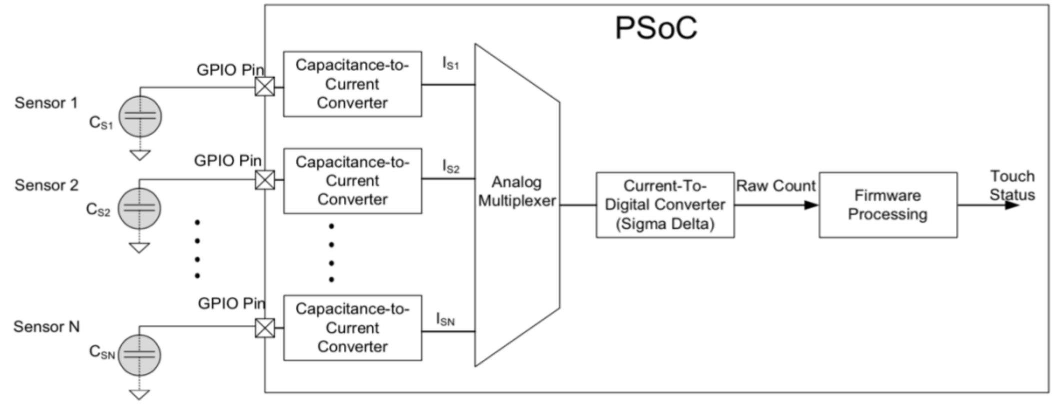
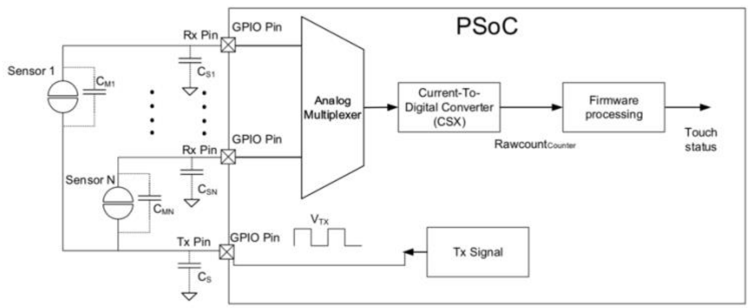
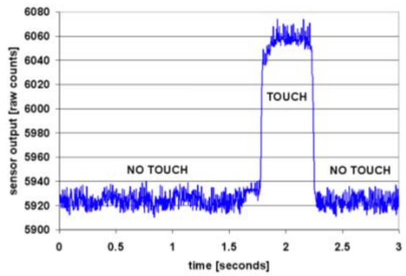


Figure 2-13 shows a plot of raw count over time. When a finger touches the sensor, the C_S increases from C_P to $C_P + C_F$, and the raw count increases. By comparing the change in raw count to a predetermined threshold, logic in firmware decides whether the sensor is active (finger is present).

https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 15.

	2.3.2 CapSense Crosspoint (CSX) Sensing Method Figure 2-12 shows the simplified block diagram of the CSX method. Figure 2-12. Simplified Diagram of CSX Method  With CSX, a voltage on the Tx pin (or Tx electrode) couples charge on to the RX pin. This charge is proportional to the mutual capacitance between the Tx and Rx electrodes. An analog multiplexer then selects one of the Rx channel and feeds it into the current to digital converter. The output count of the current to digital converter, known as $\mathbf{Rawcount_{Counter}}$, is a digital value that is proportional to the mutual-capacitance between the Rx and Tx electrodes as shown by Equation 2-4. Equation 2-4. Raw Count and Sensor Capacitance Relationship in CSX $\mathrm{Rawcount_{Counter}} = G_{CM} C_M$ Where G_{CM} is the capacitance to digital conversion gain of Mutual Capacitance method, and C_M is the mutual-capacitance between two electrodes. Figure 2-13 shows a plot of raw count over time. When a finger touches the sensor, C_M decreases from C_M to C^1_M (see Figure 2-10) hence the counter output decreases. The firmware normalizes the raw count such that the raw counts go high when C_M decreases. This is to maintain the same visual representation of raw count between CSD and CSX methods. By comparing the change in raw count to a predetermined threshold, logic in firmware decides whether the sensor is active (finger is present). Figure 2-13. Raw Count versus Time  https://www.cypress.com/documentation/application-notes/an64846-getting-started-capsense at 16.

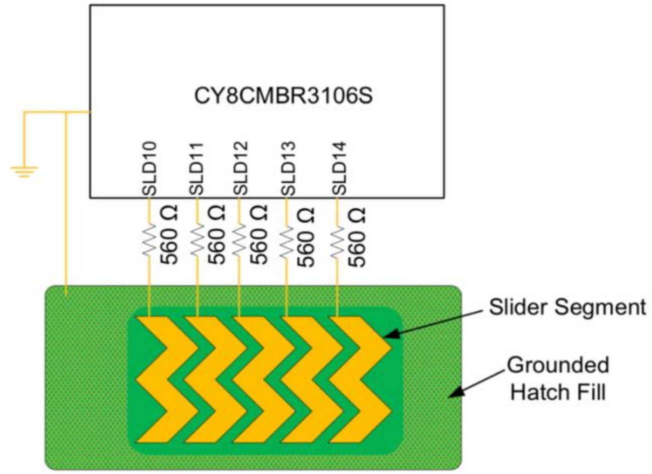
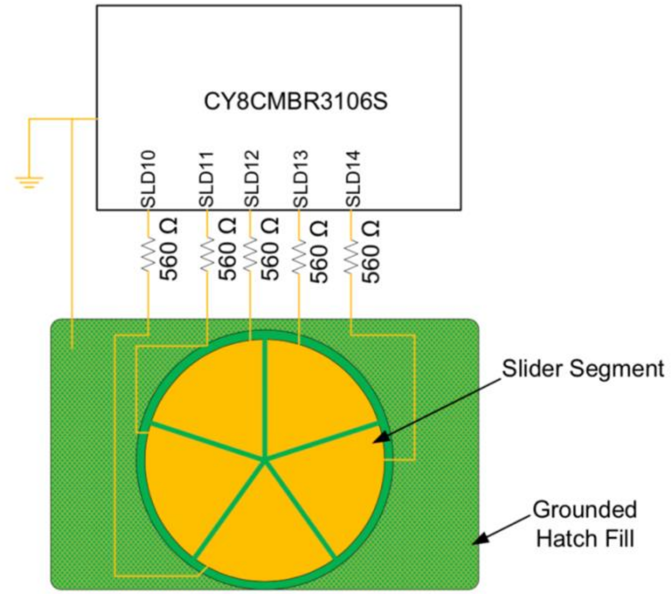
	Each slider segment connects to a pin marked for slider functionality on the CY8CMBR3106S controller. In CY8CMBR3106S, a SLDxx pin indicates that a slider segment can be connected to it. A zigzag pattern (double chevron) is recommended for slider segments. This layout ensures that when a segment is touched, the adjacent segments are also partially touched, which helps in the calculation of the centroid position. See the General Layout Guidelines section for layout recommendations. Figure 2-9. Linear Slider  Radial sliders are similar to linear sliders except that radial sliders are continuous. Figure 2-10 shows a typical radial slider. https://www.cypress.com/file/46366/download at 16.

	Figure 2-10. Radial Slider  https://www.cypress.com/file/46366/download at 16.